CLAIMS:

1. A Viterbi decoding apparatus comprising:

at least one device for allocating among plural, parallel Viterbi decoders pairs of output symbols of a convolutional encoder and for merging output of the plural decoders to form a decoded bitstream; and

the plural decoders, each configured with a trellis stage formed from two constituent trellis stages so that any path metric being updated at said stage is updated no more than once at said stage.

2. The apparatus of claim 1, wherein one of the symbols of a pair has been generated in a clock cycle of said encoder that consecutively follows a clock cycle in which the other was generated.

3. The apparatus of claim 1, wherein a decoder of the plurality is configured so that such an updated path metric has been updated at said stage by a branch metric calculated using both of the symbols of a single such pair.

4. The apparatus of claim 1, wherein the constituent trellis stages are consecutive and identical.

5. The apparatus of claim 1, wherein each constituent trellis stage defines the convolutional encoder.

6. The apparatus of claim 1, configured with a total of two decoders.

7. The apparatus of claim 6, wherein said encoder inputs a single bit for each of said output symbols.

8. The apparatus of claim 1, wherein the pairs are divided among blocks, said allocating allocates the blocks to respective ones of the decoders, each block having a

beginning and an end, the end of one block overlapping, content-wise, the beginning of a next block to form corresponding overlap regions of the two blocks, said regions having at least one of said pairs in common.

9. The apparatus of claim 1, wherein the pairs are divided among non-overlapping blocks, wherein said allocating allocates the blocks to respective ones of the decoders.

10. A Viterbi decoding method comprising the steps of:

allocating among plural, parallel Viterbi decoders pairs of output symbols of a convolutional encoder;

operating the plural decoders with a trellis stage formed from two constituent trellis stages so that any path metric being updated at said stage is updated no more than once at said stage; and

merging output of the plural decoders to form a decoded bitstream.

11. The method of claim 10, wherein one of the symbols of a pair has been generated in a clock cycle of said encoder that consecutively follows a clock cycle in which the other was generated.

12. The method of claim 10, wherein the operating step further comprises the step of updating said any such path metric from a single, respective branch metric derived from both of the output symbols of a pair of said pairs.

13. The method of claim 10, wherein the constituent trellis stages are consecutive and identical.

14. The method of claim 10, wherein each constituent trellis stage defines the convolutional encoder.

15. The method of claim 10, wherein the plurality of decoders consists in total of two decoders.

16. The method of claim 15, wherein said encoder inputs a single bit for each of said output symbols.

17. A method for testing a system that includes the plural decoders of claim 10 using the method of claim 10, wherein a component of the system is capable of operating at a higher bandwidth than a decoder of the plural decoders, further comprising the steps of:

providing said system; and

operating said system using said Viterbi decoding method, said higher bandwidth being accommodated by concurrent performance of the plural decoders.

18. The method of claim 17, wherein the component is disposed upstream of the plural Viterbi decoders.

19. The method claim 10, wherein the allocating step includes the step of dividing the pairs among blocks so that said allocating allocates the blocks to respective ones of the decoders, each block having a beginning and an end, the end of one block overlapping, content-wise, the beginning of a next block to form corresponding overlap regions of the two blocks, said regions having at least one of said pairs in common.

20. The method of claim 10, wherein the allocating step includes the step of dividing the pairs among non-overlapping blocks so that said allocating allocates the blocks to respective ones of the decoders.